

Critical density of activated random walks on \mathbb{Z}^d and general graphs

Alexandre Stauffer*

Lorenzo Taggi†

Abstract

We consider the activated random walk model on general vertex-transitive graphs. A central question in this model is whether the critical density μ_c for sustained activity is strictly between 0 and 1. It was known that $\mu_c > 0$ on \mathbb{Z}^d , $d \geq 1$, and that $\mu_c < 1$ on \mathbb{Z} for small enough sleeping rate. We show that $\mu_c \rightarrow 0$ as $\lambda \rightarrow 0$ in all transient graphs, implying that $\mu_c < 1$ for small enough sleeping rate. We also show that $\mu_c < 1$ for any sleeping rate in any graph in which simple random walk has positive speed. Furthermore, we prove that $\mu_c > 0$ in any amenable graph, and that $\mu_c \in (0, 1)$ for any sleeping rate on regular trees.

1 Introduction

In this paper we consider the activated random walk (ARW) model on a vertex-transitive graph $G = (V, E)$. This is a continuous-time interacting particle system with conserved number of particles, where each particle can be in one of two states: A (active) or S (inactive, sleeping). Initially, the number of particles at each vertex of G is an independent Bernoulli random variable of parameter $\mu \in (0, 1]$, usually called the *particle density*, and all particles are of type A. Each A-particle performs an independent, continuous time random walk on G with jump rate 1, and with each jump being to a uniformly random neighbor. Moreover, every A-particle has a Poisson clock of rate $\lambda > 0$ (called the *sleeping rate*). When the clock of a particle rings, if the particle does not share the site with other particles, the transition $A \rightarrow S$ occurs (that is, the particle becomes of type S); otherwise nothing happens. Each S-particle does not move and remains sleeping until another particle jumps into its location. At such an instant, the S-particle turns into type A, giving the transition $A+S \rightarrow 2A$.

For any given λ , it is expected that ARW undergoes a phase transition as μ varies. For example, if μ is very small, there is a lot of empty space between particles, which allows each particle to eventually fall asleep (that is, turn into type S) and never become active again. When this happens, we say that *ARW fixates*. When this does not happen, we say that *ARW is active*. This case is expected to occur when μ is large, since active particles will repetitively jump on top of other particles, “waking up” the ones that had turned into type S.

In a seminal paper, Rolla and Sidoravicius [6] showed that this process satisfies a 0-1 law (i.e., the process is either active or fixated with probability 1) and is monotone with respect to μ . This gives the existence of a critical value

$$\mu_c = \mu_c(\lambda) := \inf \{ \mu \geq 0 : P(\text{ARW is active}) > 0 \} \quad (1)$$

*University of Bath, Bath, UK; a.stauffer@bath.ac.uk. Supported in part by a Marie Curie Career Integration Grant PCIG13-GA-2013-618588 DSRELIS.

†Technische Universität Darmstadt, Darmstadt, DE; taggi@mathematik.tu-darmstadt.de. Supported by German Research Foundation (DFG).

such that ARW is active almost surely for all $\mu > \mu_c$, and fixates almost surely for all $\mu < \mu_c$. Though [6], as well as almost all existing works, are restricted to the case of G being \mathbb{Z}^d , the above properties hold for any vertex-transitive graph. Throughout this paper we always consider that G is an infinite graph that is locally finite and vertex transitive, which ensures the existence of μ_c .

Our definition above implies that $\mu_c \leq 1$ since particles are initially distributed as Bernoulli random variables. However, even if we replace this with any product measure of density $\mu > 0$, it is intuitive that $\mu_c \leq 1$, since at most one particle can fall asleep at any given vertex. This has been established for a large class of graphs [6, 8, 1]. A fundamental and very important problem in activated random walks [6, 3] is whether

$$\mu_c \in (0, 1) \text{ for all } \lambda > 0 \text{ and all vertex-transitive graphs.} \quad (2)$$

This problem is widely open, and both sides of the above question (i.e., whether $\mu_c > 0$ or $\mu_c < 1$) turned out to be very challenging. In fact, even showing that $\mu_c < 1$ for *some* $\lambda > 0$ is already quite difficult, and is only known to hold on \mathbb{Z} , thanks to a very recent paper by Basu, Ganguly and Hoffman [2]. Our first theorem establishes this result in all *transient* graphs, which includes \mathbb{Z}^d , $d \geq 3$.

Theorem 1.1. *For any vertex-transitive, transient graph, it holds that*

$$\mu_c \rightarrow 0 \text{ as } \lambda \rightarrow 0.$$

In particular, we have that $\lim_{\lambda \downarrow 0} \frac{\mu_c}{\lambda^{1/4}} < \infty$.

Regarding the question of whether $\mu_c < 1$ for *all* $\lambda > 0$, this has until now not been established for any single graph. A positive answer to this question has only been given for a variant of ARW where particles move according to *biased* random walks on \mathbb{Z}^d ; see Taggi [10]. Rolla and Tournier [7] further extended this result by proving that, for biased ARW on \mathbb{Z}^d , we have $\mu_c(\lambda) \rightarrow 0$ as $\lambda \rightarrow 0$. In our second theorem we give a positive answer to this question for the original, unbiased model, and for all graphs where simple random walk has positive speed. If $(X_t)_{t \in \mathbb{N}}$ is a random walk on G starting from a vertex x , and $|X_t|$ denotes the distance between X_t and x , we say that a random walk on G has *positive speed* if $\liminf_{t \rightarrow \infty} \frac{|X_t|}{t} > 0$ almost surely. This includes, for example, all non-amenable graphs that are vertex transitive.

Theorem 1.2. *For any vertex-transitive graph such that a random walk on it has positive speed α , it holds that*

$$\mu_c < 1 \quad \text{for all } \lambda > 0.$$

In particular, we obtain that $\mu_c < 1 - \frac{\alpha\delta}{1+\lambda}$, where δ is the probability that a random walk does not return to the origin.

We prove the theorem above by providing general sufficient conditions for ARW to be active, which as a consequence establishes an upper bound on μ_c . We believe this result is of independent interest and state it in Theorem 5.1.

For the other side of (2) (i.e., whether $\mu_c > 0$), there has been a bit more progress. It has been settled when G is \mathbb{Z}^d thanks to the seminal work of Rolla and Sidoravicius [6] for $d = 1$, and an elaborate proof of Sidoravicius and Teixeira [9] for $d \geq 2$. Our next theorem establishes that $\mu_c > 0$ in any *amenable* graph, which includes \mathbb{Z}^d , $d \geq 1$. We remark that not only our result generalizes the ones in [6, 9], but also provides the additional information that $\mu_c \rightarrow 1$ as $\lambda \rightarrow \infty$. In addition, our proof is quite short in comparison to [9].

Theorem 1.3. *For any vertex-transitive, amenable graph, we have*

$$\mu_c > 0 \quad \text{for all } \lambda > 0.$$

In particular, we have $\mu_c \geq \frac{\lambda}{1+\lambda}$.

Remark 1.4. Our lower bound is sharp, in the sense that there are no better lower bounds for μ_c which are just a function of λ and hold for any amenable graph and any jump distribution. Indeed, μ_c is known to be equal to $\frac{\lambda}{1+\lambda}$ on \mathbb{Z} with totally asymmetric jumps [5].

Remark 1.5. Our Theorems 1.1 and 1.3 hold in more generality, for any distribution of the initial location of the particles and for any jump distribution (biased or unbiased) which is translation invariant and has finite support.

Note that Theorems 1.2 and 1.3 provide a final answer to (2) in graphs that are amenable but for which a random walk has positive speed; for example, the so-called lamplighter graphs. In our final result, we also establish (2) for the case of regular trees.

Theorem 1.6. *When G is a regular tree, we have*

$$\mu_c \in (0, 1) \quad \text{for all } \lambda > 0.$$

In addition, we have $\mu_c \rightarrow 0$ as $\lambda \rightarrow 0$.

We now give a brief description of our proof techniques. The traditional strategy to establish bounds on μ_c is to consider a ball $B_L \subset V$ of some large radius L , centered at a given vertex $x \in V$, and *stabilize* ARW inside this ball. This consists of letting the process run (i.e., particles move and fall asleep) inside B_L , deleting every particle that exits B_L . This procedure will eventually end. At this point, each vertex of B_L will either contain a sleeping particle or contain no particle; such a vertex is usually called *stable*. It was shown in [6] that, roughly speaking, ARW is active if and only if the number of times particles visit x during the stabilization of B_L goes to infinity with L . In this paper, we introduce a new point of view on such stabilization procedure by focusing on some vertex $y \in B_L$, and carrying out what we call a *weak stabilization of B_L with respect to y* . Intuitively, in the weak stabilization we perform the steps of a stabilization procedure until each vertex of $B_L \setminus \{y\}$ is stable while y is allowed to be either stable or host exactly one active particle. This strategy allows us to estimate the probability that, at the end of a stabilization procedure, y contains a sleeping particle. In principle, the density of sleeping particles should correspond to μ_c , and it is by controlling such probability that we obtain estimates on μ_c . We believe that our weak stabilization procedure and our point of view of estimating the density of sleeping particles have the potential to foster even more substantial progress in this model. In particular, we believe our estimate on the probability that a sleeping particle ends at some vertex is of independent interest, and we state it in Theorem 3.1.

The remaining of the paper is organized as follows. In Section 2, we describe the so-called Diaconis-Fulton representation of ARW and its properties, which we employ in all of our proofs. Then, in Section 3, we introduce the weak stabilization procedure and estimate the probability of having a sleeping particle at a given vertex (Theorem 3.1). Next we turn to the proofs of our main results: we prove Theorem 1.1 in Section 4, Theorem 1.2 in Section 5, Theorem 1.3 in Section 6, and Theorem 1.6 in Section 7.

2 Diaconis-Fulton representation

In this section we describe the Diaconis-Fulton graphical representation for the dynamics of ARW, following [6]. For a graph $G = (V, E)$, the state of configurations is $\Omega = \{0, \rho, 1, 2, 3, \dots\}^V$,

where a vertex being in state ρ denotes that the vertex has one sleeping particle, while being in state $i \in \{0, 1, 2, \dots\}$ denotes that the vertex contains i active particles. We employ the following order on the states of a vertex: $0 < \rho < 1 < 2 < \dots$. In a configuration $\eta \in \Omega$, a site $x \in V$ is called *stable* if $\eta(x) \in \{0, \rho\}$, and it is called *unstable* if $\eta(x) \geq 1$. We fix an array of *instructions* $\tau = (\tau^{x,j} : x \in V, j \in \mathbb{N})$, where $\tau^{x,j}$ can either be of the form τ_{xy} or $\tau_{x\rho}$. We let τ_{xy} with $x, y \in V$ denote the instruction that a particle from x jumps to vertex y , and $\tau_{x\rho}$ denote the instruction that a particle from x falls asleep. Henceforth we call τ_{xy} a *jump instruction* and $\tau_{x\rho}$ a *sleep instruction*. Therefore, given any configuration η , performing the instruction τ_{xy} in η yields another configuration η' such that $\eta'(z) = \eta(z)$ for all $z \in V \setminus \{x, y\}$, $\eta'(x) = \eta(x) - \mathbf{1}(\eta(x) \geq 1)$, and $\eta'(y) = \eta(y) + \mathbf{1}(\eta(x) \geq 1)$. We use the convention that $1 + \rho = 2$. Similarly, performing the instruction $\tau_{x\rho}$ to η yields a configuration η' such that $\eta'(z) = \eta(z)$ for all $z \in V \setminus \{x\}$, and if $\eta(x) = 1$ we have $\eta'(x) = \rho$, otherwise $\eta'(x) = \eta(x)$.

Let $h = (h(x) : x \in V)$ count the number of instructions used at each site. We say that we *use* an instruction at x (or that we *topple* x) when we act on the current particle configuration η through the operator Φ_x , which is defined as,

$$\Phi_x(\eta, h) = (\tau^{x, h(x)+1} \eta, h + \delta_x). \quad (3)$$

The operation Φ_x is *legal* for η if x is unstable in η , in which case we set $\delta_x = 1$, otherwise it is *illegal* and we set $\delta_x = 0$.

Properties. We now describe the properties of this representation. Later we discuss how they are related to the stochastic dynamics of ARW. For a sequence of vertices $\alpha = (x_1, x_2, \dots, x_k)$, we write $\Phi_\alpha = \Phi_{x_k} \Phi_{x_{k-1}} \dots \Phi_{x_1}$ and we say that Φ_α is *legal* for η if Φ_{x_ℓ} is legal for $\Phi_{(x_{\ell-1}, \dots, x_1)}(\eta, h)$ for all $\ell \in \{1, 2, \dots, k\}$. Let $m_\alpha = (m_\alpha(x) : x \in V)$ be given by, $m_\alpha(x) = \sum_\ell \mathbf{1}(x_\ell = x)$, the number of times the site x appears in α . We write $m_\alpha \geq m_\beta$ if $m_\alpha(x) \geq m_\beta(x) \forall x \in V$. Analogously we write $\eta' \geq \eta$ if $\eta'(x) \geq \eta(x)$ for all $x \in V$. We also write $(\eta', h') \geq (\eta, h)$ if $\eta' \geq \eta$ and $h' = h$.

Let η, η' be two configurations, x be a vertex in V and τ be a realization of the array of instructions. Let V' be a finite subset of V . A configuration η is said to be *stable* in V' if all the sites $x \in V'$ are stable. We say that α is contained in V' if all its elements are in V' , and we say that α *stabilizes* η in V' if every $x \in V'$ is stable in $\Phi_\alpha \eta$. The following lemmas give fundamental properties of the Diaconis-Fulton representation. For the proof, please refer to [6].

Lemma 2.1 (Abelian Property). *Given any $V' \subset V$, if α and β are both legal sequences for η that are contained in V' and stabilize η in V' , then $m_\alpha = m_\beta$. In particular, $\Phi_\alpha \eta = \Phi_\beta \eta$.*

For any subset $V' \subset V$, any $x \in V'$, any particle configuration η , and any array of instructions τ , we denote by $m_{V', \eta, \tau}(x)$ the number of times that x is toppled in the stabilization of V' starting from configuration η and using the instructions in τ . Note that by Lemma 2.1, we have that $m_{V', \eta, \tau}$ is well defined.

Lemma 2.2 (Monotonicity). *If $V' \subset V'' \subset V$ and $\eta \leq \eta'$, then $m_{V', \eta, \tau} \leq m_{V'', \eta', \tau}$.*

By monotonicity, given any growing sequence of subsets $V_1 \subseteq V_2 \subseteq V_3 \subseteq \dots \subseteq V$ such that $\lim_{m \rightarrow \infty} V_m = V$, the limit

$$m_{\eta, \tau} = \lim_{m \rightarrow \infty} m_{V_m, \eta, \tau},$$

exists and does not depend on the particular sequence $\{V_m\}_m$.

We now introduce a probability measure on the space of instructions and of particle configurations. We denote by \mathcal{P} the probability measure according to which, for any $x \in V$ and any

$j \in \mathbb{N}$, $\mathcal{P}(\tau^{x,j} = \tau_{x\rho}) = \frac{\lambda}{1+\lambda}$ and $\mathcal{P}(\tau^{x,j} = \tau_{xy}) = \frac{1}{d(1+\lambda)}$ for any $y \in V$ neighboring x , where d is the degree of each vertex of G and the $\tau^{x,j}$ are independent across different values of x or j . Finally we denote by $P^\nu = \mathcal{P} \otimes \nu$ the joint law of η and τ , where ν is a distribution on Ω giving the law of η . Let \mathbb{P}^ν denotes the probability measure induced by the ARW process when the initial distribution of particles is given by ν . We shall often omit the dependence on ν by writing P and \mathbb{P} instead of P^ν and \mathbb{P}^ν . The following lemma relates the dynamics of ARW to the stability property of the representation.

Lemma 2.3 (0-1 law). *Let ν be a translation-invariant, ergodic distribution with finite density. Let $x \in V$ be any given vertex of G . Then $\mathbb{P}^\nu(\text{ARW fixates}) = P^\nu(m_{\eta,\tau}(x) < \infty) \in \{0, 1\}$.*

Roughly speaking, the next lemma gives that removing an instruction sleep, cannot decrease the number of instructions used at a given vertex for stabilization. In order to state the lemma, consider an additional instruction ι besides τ_{xy} and $\tau_{x\rho}$. The effect of ι is to leave the configuration unchanged; i.e., $\iota\eta = \eta$. Then given two arrays $\tau = (\tau^{x,j})_{x,j}$ and $\tilde{\tau} = (\tilde{\tau}^{x,j})_{x,j}$, we write $\tau \leq \tilde{\tau}$ if for every $x \in V$ and $j \in \mathbb{N}$, we either have $\tilde{\tau}^{x,j} = \tau^{x,j}$ or we have $\tilde{\tau}^{x,j} = \iota$ and $\tau^{x,j} = \tau_{x\rho}$.

Lemma 2.4 (Monotonicity with enforced activation). *Let τ and $\tilde{\tau}$ be two arrays of instructions such that $\tau \leq \tilde{\tau}$. Then, for any finite subset $V' \subset V$ and configuration $\eta \in \Omega$, we have $m_{V',\eta,\tau} \leq m_{V',\eta,\tilde{\tau}}$.*

When we average over η and τ using the measure P , we will simply write $m_{V'}$ instead of $m_{V',\eta,\tau}$.

3 Weak stabilization

In this section we introduce our method of weak stabilization and use it to derive upper and lower bounds on the probability that a given vertex contains an S -particle at the end of the stabilization of some set. This is the content of Theorem 3.1 below, which will play a fundamental role in the proofs of our main results. For any finite set $K \subset V$ and any vertex $x \in V$, let $Q(x, K)$ be the probability that there is one S -particle at x at the end of the stabilization of K .

Theorem 3.1. *Consider ARW on a vertex-transitive graph $G = (V, E)$. Then, for any $K \subset V$ and any $x \in K$, we have*

$$Q(x, K) \geq \frac{\lambda}{1+\lambda} P(m_K(x) \geq 1). \quad (4)$$

Moreover, if G is a transient graph, then

$$Q(x, K) \leq 2\sqrt{\lambda(C_G(1+\lambda) + 1)}, \quad (5)$$

where C_G is the expected number of times a simple random walk on G starting from x visits x .

In the proof of the theorem above we will employ the notion of *weakly stable configurations* and *weak stabilization*.

Definition 3.2 (weakly stable configurations). We say that a configuration η is *weakly stable* in a subset $K \subset V$ with respect to a vertex $x \in K$ if $\eta(x) \leq 1$ and $\eta(y) \leq \rho$ for all $y \in K \setminus \{x\}$. In words, this means that all vertices in $K \setminus \{x\}$ are stable, and x is either stable or hosts at most one active particle. For conciseness, we just write that η is weakly stable for (x, K) .

Definition 3.3 (weak stabilization). Given a subset $K \subset V$ and a vertex $x \in K$, the *weak stabilization* of (x, K) is a sequence of topplings of unstable sites of $K \setminus \{x\}$ and of topplings of x whenever x has at least two active particles, until a weakly stable configuration for (x, K) is obtained. The order of the topplings of a weak stabilization can be arbitrary.

The main idea of the proof of Theorem 3.1 is to perform a certain sequence of topplings to stabilize K that will allow us to control whether there is a sleeping particle at x . From the Abelian property (Lemma 2.1), in order to stabilize K we can perform the topplings in any order we want. We will stabilize K by first weakly stabilizing (x, K) , which gives a weakly stable configuration η_1 for (x, K) . Then either η_1 is stable for K , in which case we finish the stabilization procedure, or $\eta_1(x) = 1$. In the latter case, we topple x and weakly stabilize (x, K) again, obtaining a configuration η_2 . We repeat the above procedure until we obtain a stable configuration for K , concluding the stabilization. We will refer to this stabilization procedure as a *stabilization via weak stabilization*.

3.1 Proof of the lower bound in Theorem 3.1

Note that, in a stabilization via weak stabilization, after each weakly stable configuration η_i we obtain, if η_i is not stable, then with probability $\frac{\lambda}{1+\lambda}$ we encounter an instruction sleep at x , transforming η_i into a stable configuration. With this we can derive the lower bound (4) in Theorem 3.1.

Proof of (4) in Theorem 3.1. We apply the stabilization of K via weak stabilizations of (x, K) . Let η_1 be the first weakly stable configuration for (x, K) that is obtained in this procedure. As discussed above, if η_1 is not stable for K , then we obtain a stable configuration for K if the next instruction at x is sleep. Hence,

$$Q(x, K) \geq P(\eta_1 \text{ is not stable for } K) \frac{\lambda}{1 + \lambda}.$$

The proof is concluded by noting that the event that η_1 is not stable for K is equivalent to the event that x is toppled at least once. This is true because of the following. If η_1 is not stable for K , then $\eta_1(x) = 1$ which implies that x will be toppled at least once. In the other direction, if x is toppled at least once, then this happens either before η_1 is obtained or because $\eta_1(x) = 1$. But if x was toppled before η_1 was obtained, this must have happened at a time when x had at least two particles. From this time onwards, x will have at least one active particle until η_1 is obtained. Hence, η_1 is not stable. \square

3.2 Proof of the upper bound in Theorem 3.1

Our proof of the upper bound (5) for $Q(x, K)$ is a bit longer than the proof of the lower bound. We will perform the stabilization of K via weak stabilization as described above. The idea is to estimate the probability that, for any $i \geq 1$, we obtain a stable configuration for K after the i th weak stabilization of (x, K) . We do this by relating this probability to the probability that a random walk starting from x never returns to x . It is at this step that we use that G is transient.

After the i th time we perform the weak stabilization of (x, K) , we let $m_{(x, K)}^i(y)$ be the number of instructions that have been used at $y \in K$ up to this time, and denote by η_i the configuration we then obtained. Also, let $T_{(x, K)}$ denote the number of weak stabilizations of (x, K) we perform until a stable configuration in K is obtained. Note that $\eta_{T_{(x, K)}}$ is either a stable configuration, which implies that $\eta_{T_{(x, K)}}(x) = 0$, or $\eta_{T_{(x, K)}}$ is weakly stable for (x, K) with $\eta_{T_{(x, K)}}(x) = 1$ and

the next instruction used at x was a sleep instruction, thereby concluding the stabilization of K . For consistency, for any $i > T_{(x,K)}$, let η_i be the stable configuration obtained after stabilizing K and, for any $y \in K$, define $m_{(x,K)}^i(y) = m_K(y)$, which is the total number of instructions used at y for the complete stabilization of K . By the Abelian property, the quantities $T_{(x,K)}$ and $m_{(x,K)}^i$ are all well defined.

Below we state a lemma, and then show how this lemma implies the upper bound on $Q(x, K)$.

Lemma 3.4. *Given any vertex-transitive, transient graph $G = (V, E)$, any subset $K \subset V$ and any vertex $x \in K$, and letting C_G be the expected number of visits to x of a random walk on G starting from x , we have*

$$E[T_{(x,K)}] \leq C_G(1 + \lambda) + 1,$$

where the expectation is with respect to the measure P .

Proof of (5) in Theorem 3.1. For simplicity, write $\eta' = \eta_{T_{(x,K)}+1}$ for the configuration obtained after complete stabilization of K . Then the following expression holds, as the sum is over disjoint events,

$$Q(x, K) = P(\eta'(x) = \rho) = \sum_{k=1}^{\infty} P(T_{(x,K)} = k, \eta'(x) = \rho). \quad (6)$$

Now observe that

$$P(T_{(x,K)} = k, \eta'(x) = \rho) \leq \left(\frac{1}{1+\lambda}\right)^{k-1} \frac{\lambda}{1+\lambda}. \quad (7)$$

The previous inequality follows from independence of instructions: the event in the left-hand side implies that after each weak stabilization we have an active particle at x , and moreover we encounter a jump instruction at x after each of the first $k-1$ weak stabilizations, and a sleep instruction at x after the last weak stabilization. Hence, for any $H \geq 1$ we can write

$$\begin{aligned} Q(x, K) &\leq \frac{\lambda}{1+\lambda} \sum_{k=1}^H \left(\frac{1}{1+\lambda}\right)^{k-1} + P(T_{(x,K)} > H) \\ &\leq 1 - \left(\frac{1}{1+\lambda}\right)^H + \frac{E[T_{(x,K)}]}{H}, \end{aligned}$$

where in the last step we used Markov's inequality. From Lemma 3.4, we obtain

$$\begin{aligned} Q(x, K) &\leq 1 - \left(\frac{1}{1+\lambda}\right)^H + \frac{C_G(1+\lambda) + 1}{H} \\ &\leq 1 - (1-\lambda)^H + \frac{C_G(1+\lambda) + 1}{H} \\ &\leq \lambda H + \frac{C_G(1+\lambda) + 1}{H}. \end{aligned}$$

Setting $H = \sqrt{\frac{C_G(1+\lambda)+1}{\lambda}}$, which optimizes the above bound, we get

$$Q(x, K) \leq 2\sqrt{\lambda(C_G(1+\lambda) + 1)}.$$

Note that the calculations above rely on the value of H being at least 1. On the other hand, when our choice of H is smaller than 1, which implies that $\lambda > C_G(1+\lambda) + 1$, the right-hand side above is at least $2(C_G(1+\lambda) + 1) > 1$, where in the last step we use that C_G is at least 1. So our bound for $Q(x, K)$ is anyway correct. \square

3.3 Proof of Lemma 3.4

In this section we establish the upper bound on $E[T_{(x,K)}]$ from Lemma 3.4. Let $x \in V$ be a given vertex. Let E^x denote the expectation E conditioned on the initial particle configuration having one active particle at x , and E_x denote the expectation conditioned on the initial particle configuration having no particle at x .

Lemma 3.5. *For any subset $K \subset V$ and vertex $x \in K$ we have*

$$E_x[m_K(x)] \leq E^x[m_{(x,K)}^1(x)].$$

Proof. Consider an initial particle configuration η having no particle at x , and the particle configuration η^x obtained from η by adding an active particle at x . We will show a stronger result saying that, by using the same instruction array for both η and η^x , $m_K(x)$ starting from η is at most $m_{(x,K)}^1(x)$ starting from η^x . We stabilize K starting from η via weak stabilization of (x, K) , and do the same topplings for η^x . Since η and η^x differ only at x , until the first weak stabilization of η is concluded, the same topplings can be carried out in η^x as well. At this point, if there is a particle at x in η , there are two particles at x in η^x . Then if the next instruction at x is a jump instruction, we can perform the same toppling in η and η^x , and we repeat this procedure until another weakly stable configuration is obtained in η . On the other hand, if the next instruction at x is a sleep instruction, then the stabilization of η is concluded, but the weak stabilization of η^x continues. Finally, if there is no particle at x at the end of a weak stabilization of η , then the stabilization of η and the weak stabilization of η^x are concluded. Therefore, under this coupling, the weak stabilization of η concludes no later than that of η^x , concluding the proof. \square

Proof of Lemma 3.4. The crucial observation is the following. Assume that $T_{(x,K)} \geq 2$. After each of the first $T_{(x,K)} - 1$ weak stabilizations of (x, K) , we must perform at least one toppling at x , and this toppling happens after the first weak stabilization of (x, K) , so it is not counted in $m_{(x,K)}^1(x)$. This gives that

$$T_{(x,K)} - 1 \leq m_K(x) - m_{(x,K)}^1(x). \quad (8)$$

The above bound also holds when $T_{(x,K)} = 1$ since $m_K(x) \geq m_{(x,K)}^1(x)$. Then the lemma follows by claiming that

$$E[m_K(x)] \leq E[m_{(x,K)}^1(x)] + C_G(1 + \lambda). \quad (9)$$

First we prove (9) with E replaced with E^x . Denote the particle that starts at x by z . From Lemma 2.4, we have that if we ignore some sleep instructions during the stabilization of K (i.e., we replace some sleep instructions in the instruction array τ with neutral instructions ι), the value of $m_K(x)$ can only increase. Therefore, we can bound $m_K(x)$ from above by carrying out a two-step stabilization procedure. In the first step, we move z ignoring any instruction sleep seen until z exits K . Then, in the second step, we stabilize K in an arbitrary manner. The expected number of topplings at x in the first step is $C_G(1 + \lambda)$, as every time the particle visits x , we find a geometrically distributed number of sleep instructions (which are replaced by instructions ι) before the particle jumps out of x . The expected number of sleep instructions found at x after every visit is $1 + \lambda$. With this we obtain

$$E^x[m_K(x)] \leq C_G(1 + \lambda) + E_x[m_K(x)] \leq C_G(1 + \lambda) + E^x[m_{(x,K)}^1(x)],$$

where the last step follows from Lemma 3.5.

Now we establish (9) with E replaced with E_x . Using Lemma 3.5, we have

$$E_x[m_K(x)] \leq E^x[m_{(x,K)}^1(x)].$$

Now for the term $E^x[m_{(x,K)}^1(x)]$, let z be the particle that starts at x . We carry out the weak stabilization of (x, K) until either it ends or x becomes with two active particles. If the end of the weak stabilization of (x, K) happens first, the same topplings would make the weak stabilization of (x, K) under E_x end. This would give that, under this event, $E^x[m_{(x,K)}^1(x)]$ and $E_x[m_{(x,K)}^1(x)]$ are equal. If x ever becomes with two active particles, then at this moment we only topple the sites containing z , ignoring all sleep instructions and moving z until it exits K . After that we carry out all topplings necessary to conclude the weak stabilization of (x, K) . Note that, with the exception of all topplings carried out to move z out of K , all the other topplings are valid and would weakly stabilize (x, K) under E_x . Therefore, we have $E^x[m_{(x,K)}^1(x)] \leq C_G(1 + \lambda) + E_x[m_{(x,K)}^1(x)]$ since $C_G(1 + \lambda)$ is the number of topplings at x performed during the motion of z . \square

4 Proof of Theorem 1.1

Let L be a positive integer, and let $x \in V$ be a fixed vertex. Let B_L be the ball of radius L centered at x . For any $y \in B_L$, let p_y be the probability that a random walk starting from y visits x before exiting B_L .

Lemma 4.1. *For any vertex-transitive, transient graph, we have $\sum_{y \in B_L} p_y \rightarrow \infty$ as $L \rightarrow \infty$.*

Proof. By symmetry, p_y is equal to the probability that a random walk starting from x visits y before returning to x and before exiting B_L . Therefore, $\sum_{y \in B_L} p_y$ is the expected number of vertices visited by a random walk starting from x before returning to x and before exiting B_L . In a transient graph, this random walk has a positive probability of never returning to x , in which case it visits at least L vertices. This establishes the lemma. \square

Proof of Theorem 1.1. We will stabilize B_L and show that, for any fixed $\mu > 0$ there exists a fixed $\lambda > 0$ small enough such that the number of topplings at x goes to infinity with L . This implies that $\mu_c \rightarrow 0$ as $\lambda \rightarrow 0$.

Let η be the initial particle configuration inside B_L and let η_s be the particle configuration inside B_L obtained after stabilization of B_L . Then η_s only contains sleeping particles. For each particle of η_s , we start a so-called *ghost particle* which performs independent simple random walk steps until exiting B_L . Let W_L be the number of visits to x by particles or ghosts, and let R_L be the number of times that x was visited by ghosts. So $W_L - R_L$ is the number of topplings at x during the stabilization of B_L . Let N_0 be the number of visits to x of a random walk that starts from x and is killed upon exiting B_L . For simplicity, let $q = q(\lambda) = 2\sqrt{\lambda(C_G(1 + \lambda) + 1)}$, the upper bound in the second part of Theorem 3.1. Hence,

$$E[W_L - R_L] = \sum_{y \in B_L} (\mu - Q(y, B_L)) p_y E[N_0] \geq (\mu - q) E[N_0] \sum_{y \in B_L} p_y. \quad (10)$$

Note that N_0 is a geometric random variable and, for any transient graph, it holds that $E[N_0] < \infty$ as $L \rightarrow \infty$. Also, Lemma 4.1 gives that for any $\mu > q$, $E[W_L - R_L] \rightarrow \infty$ as $L \rightarrow \infty$. We want to show that

$$P\left(W_L - R_L \leq \frac{E[W_L - R_L]}{3}\right) \leq c < 1, \quad (11)$$

for some constant c independent of L . This implies that $\liminf_{L \rightarrow \infty} P\left(W_L - R_L > \frac{E(W_L - R_L)}{3}\right) > 0$. By the 0-1 law, we then obtain that $W_L - R_L$ goes to infinity almost surely, concluding the proof.

In order to establish (11), note that

$$\begin{aligned} & P\left(W_L - R_L \leq \frac{E[W_L - R_L]}{3}\right) \\ &= P\left(W_L - E[W_L] + \frac{E[W_L - R_L]}{3} \leq R_L - E[R_L] - \frac{E[W_L - R_L]}{3}\right) \\ &\leq P\left(|W_L - E[W_L]| \geq \frac{E[W_L - R_L]}{3}\right) + P\left(|R_L - E[R_L]| \geq \frac{E[W_L - R_L]}{3}\right). \end{aligned} \quad (12)$$

We now use Chebyshev's inequality, which gives

$$P\left(W_L - R_L \leq \frac{E[W_L - R_L]}{3}\right) \leq 9 \frac{\text{Var}(W_L)}{E^2[W_L - R_L]} + 9 \frac{\text{Var}(R_L)}{E^2[W_L - R_L]}. \quad (13)$$

We claim that

$$\lim_{L \rightarrow \infty} \frac{\text{Var}(W_L)}{E^2[W_L - R_L]} = 0, \quad (14)$$

and that for any $\mu > 0$ and any $c > 0$, there exists λ small enough such that

$$\limsup_{L \rightarrow \infty} \frac{\text{Var}(R_L)}{E^2[W_L - R_L]} \leq \frac{q}{(\mu - q)^2}. \quad (15)$$

Note that the above bound goes to 0 as $\lambda \rightarrow 0$. Putting (14) and (15) into (13) establishes (11), which concludes the proof of the theorem.

It remains to establish (14) and (15). For any 3 independent random variables A, B, C note that

$$\text{Var}(ABC) = E[A^2]E[B^2]E[C^2] - E^2[A]E^2[B]E^2[C]. \quad (16)$$

Then using independence we can write $\text{Var}(W_L) = \sum_{y \in B_L} \text{Var}(\mathbf{1}(\eta(y) = 1) I_y N_0)$, where I_y is the indicator that a random walk starting from y visits x before exiting B_L ; hence, $p_y = E[I_y]$. Now applying (16), we obtain

$$\begin{aligned} \text{Var}(W_L) &= \sum_{y \in B_L} (\mu p_y E[N_0^2] - \mu^2 p_y^2 E^2[N_0]) \\ &= \mu E[N_0^2] \sum_{y \in B_L} p_y \left(1 - \mu p_y \frac{E^2[N_0]}{E[N_0^2]}\right) \leq \mu E[N_0^2] \sum_{y \in B_L} p_y. \end{aligned}$$

Therefore, using (10),

$$\frac{\text{Var}(W_L)}{E^2(W_L - R_L)} \leq \frac{\mu E[N_0^2]}{(\mu - q)^2 E^2[N_0] \sum_{y \in B_L} p_y} \rightarrow 0,$$

since $\sum_{y \in B_L} p_y \rightarrow \infty$ by Lemma 4.1, while all the other terms are bounded away from both infinity and zero.

Now we turn to (15). For $y \in B_L$, write

$$S_y = \mathbf{1}(\eta_s(y)), \quad s_y = E[S_y] = Q(y, B_L), \quad \text{and} \quad s_{x,y} = E[S_x S_y].$$

Using this notation, we have $R_L = \sum_{y \in B_L} S_y I_y N_0$. Since $\text{Var}(R_L) = E[R_L^2] - E^2[R_L]$, we write

$$ER_L^2 = \sum_{y \in B_L} E[S_y I_y N_0^2] + \sum_{y, z \in B_L, y \neq z} E(S_y S_z I_y I_z N_0 N_0'),$$

where N_0, N'_0 are independent and identically distributed. Using independence, we have

$$ER_L^2 = \sum_{y \in B_L} s_y p_y E[N_0^2] + \sum_{y, z \in B_L, y \neq z} s_{y,z} p_y p_z E^2[N_0].$$

Hence,

$$\begin{aligned} \text{Var}(R_L) &= \sum_{y \in B_L} s_y p_y E[N_0^2] + \sum_{y, z \in B_L, y \neq z} s_{y,z} p_y p_z E^2[N_0] - \left(\sum_{y \in B_L} s_y p_y E[N_0] \right)^2 \\ &= \sum_{y \in B_L} (s_y p_y E[N_0^2] - s_y^2 p_y^2 E^2[N_0]) + \sum_{y, z \in B_L, y \neq z} (s_{y,z} - s_y s_z) p_y p_z E^2[N_0] \\ &\leq \sum_{y \in B_L} s_y p_y E[N_0^2] + \sum_{y, z \in B_L, y \neq z} (s_y - s_y s_z) p_y p_z E^2[N_0] \\ &\leq q E[N_0^2] \sum_{y \in B_L} p_y + q E^2[N_0] \sum_{y, z \in B_L, y \neq z} p_y p_z. \end{aligned}$$

Finally, we obtain

$$\begin{aligned} \frac{\text{Var}(R_L)}{E^2(W_L - R_L)} &\leq \frac{q E[N_0^2] \sum_{y \in B_L} p_y + q E^2[N_0] \sum_{y, z \in B_L, y \neq z} p_y p_z}{(\mu - q)^2 E^2[N_0] \left(\sum_{y \in B_L} p_y \right)^2} \\ &\leq \frac{q E[N_0^2]}{(\mu - q)^2 E^2[N_0] \sum_{y \in B_L} p_y} + \frac{q}{(\mu - q)^2}. \end{aligned}$$

Note that for any fixed $\lambda > 0$ the first fraction goes to 0 with L since $\sum_x p_x \rightarrow \infty$ and all the other terms are bounded away from zero and infinity. The second term can be made arbitrarily small since $q \rightarrow 0$ as $\lambda \rightarrow 0$. In particular, if $\mu > q + \sqrt{q}$, the second term is smaller than 1, so ARW is active almost surely. This establishes (15). \square

5 Proof of Theorem 1.2

We prove Theorem 1.2 by first establishing general sufficient conditions that give $\mu_c < 1$ (Theorem 5.1 below), and then showing that graphs of positive speed for random walks satisfy those conditions. Let $x \in V$ be a fixed vertex of G , which we refer to as the *origin*. Let $\{X(t)\}_{t \in \mathbb{N}}$ denote a simple random walk on G starting from the origin, and let $\{Y(t)\}_{t \in \mathbb{N}}$ be independent random variables such that, for any $t \in \mathbb{N}$, we have $Y(t) = 0$ with probability $\frac{1}{1+\lambda}$ and $Y(t) = 1$ with probability $\frac{\lambda}{1+\lambda}$. Let B_L be the ball of radius L centered at x , and let A_L be the vertices at distance L from x . For any set $V' \subset V$, let

$$\tau_{V'} := \min\{t \in \mathbb{N} : X(t) \in V'\}$$

be the first hitting time of the random walk to V' and

$$\tau_{V'}^+ := \min\{t \geq 1 : X(t) \in V'\}$$

be the first return time of the random walk to V' . Finally, let

$$\tau_{V'}^k := \min\{t \geq 1 : X(t) \notin V' \text{ and } Y(t) = 1\}.$$

We can interpret the above quantity by considering that the random walk is “killed” outside V' at times t when $Y(t) = 1$; using this, $\tau_{V'}^k$ gives the time the random walk is killed.

Here we consider that the initial particle configuration, denoted by η , is given by any product measure on \mathbb{N}^V with density $E[\eta(x)] = \mu$. Let $\nu_0(\mu) = P(\eta(x) = 0)$.

Theorem 5.1. *Given positive integers $n < L$, set $\Lambda = V \setminus B_{\text{exp}(L)}$ and let*

$$N_n^L := |\{t \in \mathbb{N} : X(t) \in A_n \text{ and } t < \tau_\Lambda \wedge \tau_0^+\}|$$

be the number of visits of $X(t)$ to A_n before $X(t)$ enters Λ or returns to the origin. Let

$$\tilde{N}_n^L := |\{t \in \mathbb{N} : X(t) \in A_n \text{ and } t < \tau_\Lambda \wedge \tau_0^+ \wedge \tau_{B_{n-1}}^k\}|$$

be the number of visits of $X(t)$ to A_n before $X(t)$ enters Λ , returns to the origin or is “killed” outside B_{n-1} . Let also $M_L := \sum_{n=0}^L N_n^L$ and $\tilde{M}_L := \sum_{n=0}^L \tilde{N}_n^L$. If given μ and λ we have

$$\liminf_{L \rightarrow \infty} \frac{E[\tilde{M}_L]}{E[M_L]} > \frac{\nu_0(\mu)}{\mu + \nu_0(\mu)} \quad \text{and} \quad \lim_{L \rightarrow \infty} E[\tilde{M}_L] = \infty, \quad (17)$$

then ARW is active almost surely.

Proof. We will define a stabilization procedure for B_L and show that the number of topplings at the origin goes to infinity with L . We will do the stabilization by moving particles in levels, starting from $n = L$ until $n = 0$. At level n , we move each particle that is currently located in A_n until one of the following events occur:

1. the particle reaches the origin,
2. the particle reaches an empty site in B_{n-1} ,
3. the particle “uses” an instruction “sleep” in $\Lambda^c \setminus B_{n-1}$,
4. the particle reaches Λ .

After moving all particles at level n , we repeat the procedure above for particles located at level $n - 1$. Note that at level $n - 1$ we may find particles that were moved at level $n' \geq n$ but which stopped at some empty vertex $z \in A_{n-1}$ with $\eta(z) = 0$. Let G_L be the total number of particles that stop at the origin, and note that $m_{B_L}(x)$ stochastically dominates G_L . Our goal is to prove that there exists a constant $c > 0$ independent of L such that $G_L > cL$ with positive probability, which implies that ARW is active by the 0-1 law.

In order to estimate G_L , we introduce ghost particles as in Section 4. At level n , we create a ghost particle every time we move a particle currently at some vertex $z \in A_n$ for which $\eta(z) = 0$ and which stops at some $y \in \Lambda^c \setminus B_{n-1}$ because of the event 3. The ghost particle is then created at y and, from this moment onwards, performs independent simple random walk steps until reaching $\Lambda \cup \{x\}$, when it then stops. Let W_L be the number of particles and ghosts visiting the origin, and let R_L be the number of ghosts visiting the origin. Then,

$$G_L \stackrel{d}{=} W_L - R_L.$$

We now estimate the terms W_L and R_L separately. For any $j \in \mathbb{N}$ and $z \in V$, let $(X^{(z,j)}(t))_{t \in \mathbb{N}}$ be an independent random walk on V starting from z and $(Y^{(z,j)}(t))_{t \in \mathbb{N}}$ be an infinite sequence of i.i.d. random variables such that $Y^{(0,0)}(0) = 1$ with probability $\frac{\lambda}{1+\lambda}$ and $Y^{(0,0)}(0) = 0$ with probability $\frac{1}{1+\lambda}$. Let $\tau_S^{(z,j)}$ be the first time the random walk $(X^{(z,j)}(t))_{t \in \mathbb{N}}$ visits the set $S \subset V$ and let us write simply $\tau_y^{(z,j)}$ if $S = \{y\}$. Then,

$$W_L \text{ stochastically dominates } \tilde{W}_L := \sum_{n=1}^L \sum_{z \in A_n} \sum_{j=1}^{\eta(z)} \mathbf{1} \left(\mathcal{A}^{(z,j)} \cap \mathcal{B}^{(z,j)} \right), \quad (18)$$

where $\mathcal{A}^{(z,j)} := \{\tau_x^{(z,j)} < \tau_\Lambda^{(z,j)}\}$, $\mathcal{B}^{(z,j)} := \{Y^{(z,j)}(t) = 0 \text{ for any } t \leq \tau_x^{(z,j)} \text{ such that } X^{(z,j)}(t) \notin B_{|z|-1}\}$, and $|z|$ denotes the distance between z and x .

The crucial observation for the estimation of R_L is that particles produce ghosts only when they are moved from sites that are empty with respect to η , and each such site can be associated to the creation of at most one ghost. Hence, if from every site $z \in B_L$ with $\eta(z) = 0$ we start a *sleeping random walk* $(X^{(z,0)}(t), Y^{(z,0)}(t))_{t \in \mathbb{N}}$ and we count \tilde{R}_L , the number of them which hit the origin before entering Λ and such that $Y(t) = 1$ somewhere in $\Lambda^c \setminus B_{|z|-1}$, we conclude that

$$\tilde{R}_L \text{ stochastically dominates } R_L. \quad (19)$$

Hence, we write,

$$\tilde{R}_L := \sum_{n=1}^L \sum_{z \in A_n} \mathbf{1}(\eta(z) = 0) \cdot \mathbf{1}(\mathcal{A}^{(z,0)} \cap \bar{\mathcal{B}}^{(z,0)}), \quad (20)$$

where for clarity we denote by $\bar{\mathcal{B}}^{(z,0)} := (\mathcal{B}^{(z,0)})^c = \{Y^{(z,0)}(t) = 1 \text{ for some } t \leq \tau_x^{(z,0)} \text{ such that } X^{(z,0)}(t) \notin B_{|z|-1}\}$ the complement of $\mathcal{B}^{(z,0)}$. As the initial particle configuration is distributed according to a product measure, from (18) and (20) it follows that,

$$\begin{aligned} E[\tilde{W}_L] - E[\tilde{R}_L] &= \sum_{n=1}^L \sum_{z \in A_n} \left[\mu \cdot P(\mathcal{A}^{(z,0)} \cap \mathcal{B}^{(z,0)}) - \nu_0(\mu) \cdot P(\mathcal{A}^{(z,0)} \cap \bar{\mathcal{B}}^{(z,0)}) \right] \\ &= \sum_{n=1}^L \sum_{z \in A_n} \left[(\mu + \nu_0(\mu)) P(\mathcal{A}^{(z,0)} \cap \mathcal{B}^{(z,0)}) - \nu_0(\mu) \cdot P(\mathcal{A}^{(z,0)}) \right]. \end{aligned} \quad (21)$$

To simplify the notation, we will henceforth drop the 0's from the superscript in the terms above. When analyzing the term $P(\mathcal{A}^z \cap \mathcal{B}^z)$, consider the last time t that the random walk starting from $z \in A_n$ visits A_n before reaching the origin. We will denote by y the vertex of A_n where the random walk is in its last visit to A_n . Hence, decomposing in y and t , we have

$$P(\mathcal{A}^z \cap \mathcal{B}^z) = \sum_{y \in A_n} \sum_{t=1}^{\infty} P(\mathcal{C}^{z,y,t} \cap \mathcal{D}^{z,t} \cap \mathcal{E}^{z,t}) \cdot P(\tau_x^y < \tau_{A_n,+}^y), \quad (22)$$

where $\mathcal{C}^{z,y,t} := \{X^z(t) = y\}$, $\mathcal{D}^{z,t} := \{Y^z(t) = 0 \text{ for any } i \leq t \text{ such that } X^z(i) \notin B_{|z|-1}\}$, $\mathcal{E}^{z,t} := \{\tau_{\{x\} \cup \Lambda}^z > t\}$, $\tau_{S,+}^z$ is the first return time of the random walk starting from z to the set $S \subset V$. Now since graph is transitive, any path of a random walk from a vertex z_1 to z_2 occurs with the same probability as the reversed path for a random walk going from z_2 to z_1 . This gives that, for $y, z \in A_n$,

$$P(\tau_x^y < \tau_{A_n,+}^y) = P(\{X^x(\tau_{A_n}^x) = y\} \cap \{\tau_{A_n}^x < \tau_{x,+}^x\}); \quad (23)$$

that is, the event $\tau_x^y < \tau_{A_n,+}^y$ is equivalent to the event that a random walk starting from x visits A_n before returning to x , and visits A_n for the first time at y . Also, for $y, z \in A_n$ and $t \in \mathbb{N}$,

$$P(\mathcal{C}^{z,y,t} \cap \mathcal{D}^{z,t} \cap \mathcal{E}^{z,t}) = P(\mathcal{C}^{y,z,t} \cap \mathcal{D}^{y,t} \cap \mathcal{E}^{y,t}). \quad (24)$$

Now plug (23) and (24) into (22), and plug the result into (21). Summing over $z \in A_n$ first and then over y and t , and using the Markov property for the random walk, we conclude that

$$\begin{aligned} \sum_{z \in A_n} P(\mathcal{A}^z \cap \mathcal{B}^z) &= \\ \sum_{y \in A_n} \sum_{t=0}^{\infty} E \left[\sum_{z \in A_n} \mathbf{1}(\mathcal{C}^{y,z,t} \cap \mathcal{D}^{y,t} \cap \mathcal{E}^{y,t}) \right] \cdot P(\{X^x(\tau_{A_n}^x) = y\} \cap \{\tau_{A_n}^x < \tau_{x,+}^x\}) &= E[\tilde{N}_n^L]. \end{aligned} \quad (25)$$

Similarly to (25), we obtain

$$\sum_{z \in A_n} P(\mathcal{A}^z) = E[N_n^L]. \quad (26)$$

Hence, plugging (25) and (26) into (21), we have

$$\begin{aligned} E[\tilde{W}_L] - E[\tilde{R}_L] &= \left(\sum_{n=0}^L (\mu + \nu_0(\mu)) E[\tilde{N}_n^L] - \nu_0(\mu) E[N_n^L] \right) \\ &= (\mu + \nu_0(\mu)) E[\tilde{M}_L] - \nu_0(\mu) E[M_L]. \end{aligned} \quad (27)$$

Thus, if the conditions in (17) are satisfied, the lower bound above diverges with L . It remains to prove that this implies that $G_L \rightarrow \infty$ with L with positive probability, which in turn implies that ARW is active almost surely by the 0-1 law (Lemma 2.3). For this, we use the same derivation as in (12) and (13), which gives that

$$\begin{aligned} P\left(\tilde{W}_L - \tilde{R}_L < \frac{E[\tilde{W}_L] - E[\tilde{R}_L]}{3}\right) &\leq 9 \frac{\text{Var}(\tilde{W}_L)}{E^2[\tilde{W}_L - \tilde{R}_L]} + 9 \frac{\text{Var}(\tilde{R}_L)}{E^2[\tilde{W}_L - \tilde{R}_L]} \\ &\leq 9 \frac{E[\tilde{W}_L]}{E^2[\tilde{W}_L - \tilde{R}_L]} + 9 \frac{E[\tilde{R}_L]}{E^2[\tilde{W}_L - \tilde{R}_L]}, \end{aligned} \quad (28)$$

where in the last step we use that $\text{Var}(\tilde{W}_L) \leq E[\tilde{W}_L]$ and $\text{Var}(\tilde{R}_L) \leq E[\tilde{R}_L]$ since \tilde{W}_L and \tilde{R}_L are defined as a sum of independent Bernoulli random variables. Note that (27) and (17) imply that

$$E[\tilde{W}_L - \tilde{R}_L] > K E[M_L] \quad \text{for some constant } K > 0 \text{ and all large enough } L. \quad (29)$$

Hence we obtain that $E[\tilde{W}_L] \geq E[\tilde{R}_L]$ for all large enough L . In addition, from the derivation of (21) and (27) we have

$$E[\tilde{W}_L] \leq (\mu + \nu_0(\mu)) E[\tilde{M}_L].$$

Using these facts, we obtain

$$E[\tilde{W}_L + \tilde{R}_L] \leq 2E[\tilde{W}_L] \leq 2(\mu + \nu_0(\mu)) E[\tilde{M}_L].$$

Plugging this into (28), and using (29), we get

$$P\left(\tilde{W}_L - \tilde{R}_L < \frac{E[\tilde{W}_L] - E[\tilde{R}_L]}{3}\right) \leq \frac{18(\mu + \nu_0(\mu)) E[\tilde{M}_L]}{K^2 E^2[M_L]} \leq \frac{18(\mu + \nu_0(\mu))}{K^2 E[M_L]}.$$

By (17), the last term converges to 0 with L . Hence, $W_L - R_L > \frac{E[W_L - R_L]}{3}$ with positive probability for all large L , concluding the proof. \square

Proof of Theorem 1.2. We show that for any $\lambda > 0$ and $\mu > 1 - \frac{\alpha\delta}{1+\lambda}$ the conditions in (17) are satisfied. Observe that, conditioning on the non-return of the random walk to the origin, \tilde{N}_n^L is stochastically larger than a random variable which takes value 1 with probability $\frac{1}{1+\lambda}$ and 0 with probability $\frac{\lambda}{1+\lambda}$, as the random walk hits A_n at least one time. Hence,

$$E[\tilde{N}_n^L] \geq \frac{\delta}{1+\lambda} \quad \text{and, consequently,} \quad E[\tilde{M}_L] \geq \frac{\delta}{1+\lambda} L. \quad (30)$$

We use that the random walk has a positive speed α ; that is,

$$\lim_{t \rightarrow \infty} \frac{|X(t)|}{t} = \alpha \quad \text{almost surely.} \quad (31)$$

This gives that, for any positive ϵ and all t large enough, $|X(t)| \geq (\alpha - \epsilon)t$ almost surely. Then if L is large enough, for any $t > \frac{L}{\alpha - \epsilon}$ we have $X(t) > L$ almost surely, which gives that

$$E[M_L] \leq \frac{L}{\alpha - \epsilon}. \quad (32)$$

Hence, we conclude that

$$\frac{E[\tilde{M}_L]}{E[M_L]} \geq \frac{\delta(\alpha - \epsilon)}{1 + \lambda}.$$

Thus, the conditions in (17) are satisfied when $\nu_0(\mu) = 1 - \mu$ as long as $\mu > 1 - \frac{\alpha\delta}{1+\lambda}$. \square

6 Proof of theorem 1.3

Proof of theorem 1.3. Since G is amenable and vertex transitive, we can take a sequence of subsets $\{V_n\}_{n \geq 1}$ of V such that $V_n \rightarrow V$ as $n \rightarrow \infty$, there exists a vertex $x \in \bigcap_{n=1}^{\infty} V_n$, and

$$\frac{|\partial V_n|}{|V_n|} \text{ is non-increasing and goes to 0 as } n \rightarrow \infty,$$

where ∂V_n denotes the external boundary of V_n ; that is, the set of vertices in $V \setminus V_n$ that have an edge incident to V_n . Let B_K be the ball of radius K centered at x , and recall that $m_{B_K}(x)$ is the number of instructions used at x to stabilize B_K . If we assume that $\mu > \mu_c$, then the 0-1 law (Lemma 2.3) implies that $\Pr(m_{B_K}(x) \geq 1) \rightarrow 1$ as $K \rightarrow \infty$. By monotonicity of this probability, for any fixed $\epsilon > 0$, we can find $K = K(\epsilon)$ large enough such that

$$\Pr(m_{B_K}(x) \geq 1) \geq 1 - \epsilon.$$

For any set $V_n \subset V$, let V_n^K be the set obtained by taking the union of balls of radius K centered at each vertex of V_n . Hence $V_n^K \supset V_n$. Let $N_{n,K}$ be the number of particles inside V_n^K prior to the stabilization of V_n^K and let $N_{n,K}^s$ be the number of sleeping particles in V_n^K after the stabilization of V_n^K . Clearly, $N_{n,K}^s \leq N_{n,K}$ almost surely. Let d denotes the degree of each vertex of G ; so B_K has at most d^K vertices. Note that

$$E[N_{n,k}] = |V_n^K| \mu \leq (|V_n| + d^K |\partial V_n|) \mu = \left(1 + d^K \frac{|\partial V_n|}{|V_n|}\right) |V_n| \mu.$$

Also, from (4) in Theorem 3.1, we have

$$E[N_{n,K}^s] \geq \sum_{y \in V_n} Q(y, V_n^K) \geq \left(\frac{\lambda}{1 + \lambda}\right) \sum_{y \in V_n} \Pr(m_{V_n^K}(y) \geq 1).$$

Since V_n^K contains a ball of radius K centered at y , by monotonicity and transitivity we obtain

$$E[N_{n,K}^s] \geq |V_n| \left(\frac{\lambda}{1 + \lambda}\right) \sum_{y \in V_n} \Pr(m_{B_K}(x) \geq 1) \geq |V_n| \left(\frac{\lambda}{1 + \lambda}\right) (1 - \epsilon).$$

Since $E[N_{n,k}] \geq E[N_{n,k}^s]$, placing the two inequalities together yields

$$\mu \geq \left(\frac{\lambda}{1 + \lambda}\right) (1 - \epsilon) \left(1 + d^K \frac{|\partial V_n|}{|V_n|}\right)^{-1}.$$

Now set $n = n(d, K)$ large enough such that $\mu \geq \left(\frac{\lambda}{1 + \lambda}\right) (1 - \epsilon)^2$. Therefore, assuming that $\mu > \mu_c$ implies that $\mu \geq \left(\frac{\lambda}{1 + \lambda}\right) (1 - \epsilon)^2$, which completes the proof since $\epsilon > 0$ is arbitrary. \square

7 Proof of Theorem 1.6

We start with a simple, well-known lemma regarding random walks on regular trees.

Lemma 7.1. *For any ℓ , let p_ℓ be the probability that a random walk starting at distance ℓ from the origin ever visits the origin. Then, for a d -regular tree we have $p_\ell = \left(\frac{1}{d-1}\right)^\ell$.*

Proof. The lemma follows by checking that if we set $p_\ell = a^\ell$ for some $a > 0$, then $a = \frac{1}{d-1}$ is the only solution in $(0, 1)$ of the recursion $a^\ell = \frac{1}{d}a^{\ell-1} + \left(\frac{d-1}{d}\right)a^{\ell+1}$. \square

Proof of Theorem 1.6. Since a simple random walk in a d -regular tree has positive speed, Theorem 1.2 gives that $\mu_c < 1$ for any $\lambda > 0$. Also, when G is a d -regular tree, for any given $\mu > 0$, the conditions in Theorem 5.1 are satisfied by setting $\lambda > 0$ small enough. This implies that $\lim_{\lambda \downarrow 0} \mu_c = 0$. It remains to show that $\mu_c > 0$ for any $\lambda > 0$.

We assume that the initial distribution of particles is given by independent Poisson random variables of mean μ . Then since any Bernoulli random variable of mean $q \in (0, 1/2)$ is stochastically dominated by a Poisson random variable of mean $2q$, and using monotonicity of ARW (cf. Lemma 2.2), we have that if for a given $\mu > 0$ ARW almost surely fixates starting from a Poisson field of particles of density μ , then ARW almost surely fixates starting from a Bernoulli field of particles of density $\mu/2$. This establishes that $\mu_c > 0$.

We will employ a beautiful stabilization procedure developed by Rolla and Sidoravicius [7] for the one-dimensional lattice \mathbb{Z} . We will need to carry out a much more delicate analysis for the case of a d -regular tree. Let x_1, x_2, \dots be the particles ordered according to their distance to the origin, with x_1 being the closest particle to the origin. Let L be an arbitrarily large integer, and consider the finite system inside B_L , the ball of radius L around the origin. Our goal is to show that with positive probability we stabilize B_L without any particle visiting the origin.

Now we describe the stabilization procedure of Rolla and Sidoravicius. Let C_0 be only the origin. We move x_1 repetitively, ignoring all sleep instructions, until it either reaches $V \setminus B_L$ or C_0 . If it reaches $V \setminus B_L$, then let $C_1 = C_0$. Otherwise, let $z_1, z_2, \dots, z_T \in V$ be the vertices visited by x_1 , and define τ to be the largest integer so that x_1 ignored a instruction *sleep* at z_τ ; if x_1 never ignored a instruction *sleep* and visit the origin we will stop and declare that the procedure failed. Set $C_1 = C_0 \cup \{z_\tau, z_{\tau+1}, \dots, z_T\}$. We see C_1 as a set of corrupted vertices after x_1 has moved. The idea is that we have corrupted the array of instructions at the vertices inside C_1 since we have looked at these instructions after the moment at which we stopped x_1 . For this reason, we cannot use instructions at these vertices as we proceed to move other particles. We then repeat the procedure above. For each $k \geq 1$, let E_k be the event that x_k sees at least one instruction *sleep* before visiting C_{k-1} . We define C_k as C_{k-1} plus the vertices visited by x_k since the last time x_k sees a sleep instruction. Our goal is to show that there exists a positive constant c so that, for all $L \geq 1$, we have

$$P\left(\bigcap_k^{n_L} E_k\right) \geq c > 0, \quad (33)$$

where n_L is the number of particles initially inside B_L . When (33) holds we have that ARW fixates almost surely.

In order to establish (33), we will define a branching process on \mathbb{N} . To avoid ambiguity we refer to the particles of the branching process as *tokens*. At step j , the branching process will have as many tokens as connected components of the graph obtained from G by removing all vertices in C_j . We will denote this graph by $G \setminus C_j$. For each connected component and each time, the position of the token of that component will give the distance between the corrupted vertices

and the closest particle of that component. Therefore, since initially $G \setminus C_0$ has d connected components, we start the branching process with d tokens, all located at position 1 since we are yet to check whether there are particles at distance 1 from the origin.

For each $\ell \geq 1$, let Y_ℓ be the set of particles at distance ℓ from the origin. For each connected component of $G \setminus C_0$, if $Y_1 = \emptyset$ or all particles in Y_1 do not reach the origin, then we move the token corresponding to that component one position forward in \mathbb{N} . Initially, we expect to move the token forward many times since μ is small. If after moving all particles in $Y_1 \cup Y_2 \cup \dots \cup Y_{m-1}$ we have that none of them reached the origin, we obtain that the branching process contains d tokens at position m .

Suppose now that, when considering the particles in Y_m , we find a particle x_j that reaches the origin. Let κ be the token corresponding to the connected component of $G \setminus C_{j-1}$ containing x_j . Let Γ_j be the set of corrupted sites created by x_j ; so $C_j = C_{j-1} \cup \Gamma_j$. We take token κ and replace it by the number of connected components created by Γ_j ; i.e., the number of tokens we add is equal to the difference between the number of connected components of $G \setminus C_j$ and $G \setminus C_{j-1}$. Note that $G \setminus C_j$ has at most $d|\Gamma_j|$ connected components more than C_{j-1} , so we have added at most $d|\Gamma_j|$ tokens in this step. The position of the tokens is determined by the distance between C_j and m . Figure 1 illustrates one example for $d = 3$ and $m = 4$. In this example, the particle x_j is moved and sees an instruction sleep for the last time in its first visit to vertex w . Here Γ_j is the set of blue vertices, C_j comprises the blue and red vertices, and κ was replaced by 3 tokens, one at position $m - 1$ and two at position $m - 2$. So tokens move forward when particles do not reach the set of corrupted vertices, otherwise they branch and move backwards.

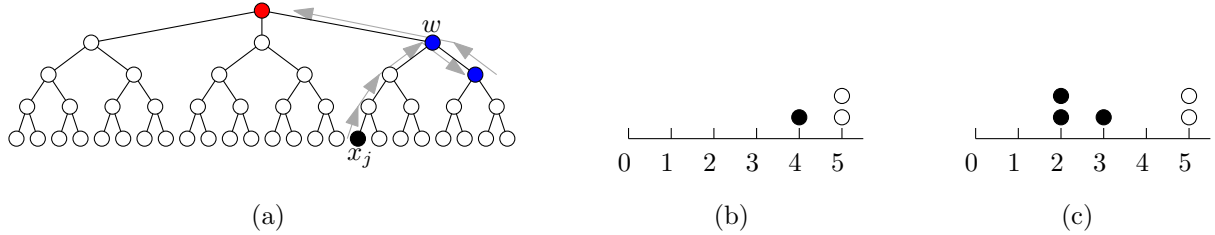


Figure 1: (a) The move of particle x_j . Gray arrows indicate the path traversed by x_j , where the last time x_j sees an instruction sleep is in its first visit to w . The red vertex represents the origin, while blue vertices represent the set of corrupted sites after the move of x_j . (b) The branching process before the move of x_j , where the black circle represents the token of the component containing x_j . Note that for all the other components, all particles at distance 4 from the origin were already moved, so their corresponding tokens are already in position 5. (c) The branching process after the move of x_j .

We show this via a Lyapunov function, which we will update in steps. Recall that we move the particles in order of their distance to the origin. When moving the particles at distance m , we will update the function at each time that we find a particle that reaches the corrupted vertices or when we have moved all the particles inside the same connected component that is at distance m from the origin. When either of these situations happen, we declare that a step of the analysis has ended and update the Lyapunov function. We will now denote by C_t the set of corrupted sites at the end of step t ; recall that C_0 contains only the origin.

Let $\alpha := \sqrt{\frac{1}{1+\lambda}}$. Given the end of a step t , and letting v_1, v_2, \dots be the positions of the tokens, define the function

$$\Psi_t = \sum_{i \geq 1} \alpha^{v_i}.$$

We assume that all vertices in the ball B_R have initially no particle, where R is chosen large enough such that $\Psi_0 = d\alpha^R < \frac{1}{5}$. This event occurs with positive probability depending on R . We observe that if any token reaches position 0 then we have $\Psi_t \geq 1$. We show that with positive probability $\Psi_t < 1$ for all $t \in \mathbb{N}$. For this it suffices to show that Ψ_t is a supermartingale, provided μ is small enough.

Let \mathcal{F}_t denote the filtration given by the position of the tokens at steps $0, 1, \dots, t$, together with the information regarding each particle that was moved in each of these steps. Assume that at step $t+1$, we need to move particles from a given connected component of $G \setminus C_t$ whose token is at position v . In other words, we will move all particles from that component that are at distance v from C_t . Note that the number of those particles that ever visit the origin is a Poisson random variable N of mean

$$p_v \mu (d-1)^{v-1} \leq \frac{\mu}{d-1},$$

where p_v is defined in Lemma 7.1. Then if $N = 0$, which happens with probability $\exp\left(-\frac{\mu}{d-1}\right)$, the token advances one position and the function Ψ_t changes to

$$\Delta_t^v = \Psi_t - \alpha^v + \alpha^{v+1}.$$

If $N \geq 1$, we will consider one of the particles that visits the origin. We need to compute the number of vertices that end up being corrupted by this particle. This is the number of vertices visited by the particle from the last time the particle sees an instruction sleep until it reaches the corrupted vertices. Since at each step the particle does not see an instruction sleep with probability $\rho = \frac{1}{1+\lambda}$, independently of everything else, we have that the number of vertices corrupted by the particle is stochastically dominated by a geometric random variable of parameter $1-\rho$. When this number is equal to i , note that the number of connected components that are created by the newly corrupted vertices is at most id . Therefore, the token is split into at most id tokens, all of them at distance at least $v-i$. The expected change of Ψ_t is then at most

$$\tilde{\Delta}_t^v = (1-\rho) \sum_{i=1}^{\infty} \rho^{i-1} (\Psi_t - \alpha^v + i d \alpha^{v-i}).$$

At this moment we end the step and update the Lyapunov function. If $N \geq 2$, we will look at the other particles in the next step. We will simply repeat the steps above, using the fact that a Poisson random variable of mean $\mu/(d-1)$ is stochastically dominated by a geometric random variable whose probability of being at least $j \geq 0$ is $(1 - \exp(-\mu/(d-1)))^j$.

Using the above strategy, we obtain that $E(\Psi_{t+1} | \mathcal{F}_t)$ is at most

$$\begin{aligned} & \exp\left(-\frac{\mu}{d-1}\right) \Delta_t^v + \left(1 - \exp\left(-\frac{\mu}{d-1}\right)\right) \tilde{\Delta}_t^v \\ &= \Psi_t - \exp\left(-\frac{\mu}{d-1}\right) (\alpha^v - \alpha^{v+1}) + \left(1 - \exp\left(-\frac{\mu}{d-1}\right)\right) (1-\rho) \sum_{i=1}^{\infty} \rho^{i-1} (-\alpha^v + i d \alpha^{v-i}) \\ &= \Psi_t - \alpha^v \exp\left(-\frac{\mu}{d-1}\right) (1-\alpha) + \alpha^{v-1} \left(1 - \exp\left(-\frac{\mu}{d-1}\right)\right) (1-\rho) \sum_{i=1}^{\infty} \rho^{i-1} (i d \alpha^{-i+1} - \alpha) \\ &\leq \Psi_t - \alpha^v \exp\left(-\frac{\mu}{d-1}\right) (1-\alpha) + \alpha^{v-1} \left(1 - \exp\left(-\frac{\mu}{d-1}\right)\right) (1-\rho) d \sum_{i=1}^{\infty} i (\rho/\alpha)^{i-1}. \end{aligned}$$

Since we set α such that $\rho/\alpha < 1$, then the sum above converges to $C = \left(\frac{\alpha}{\alpha-\rho}\right)^2$. Then

$$E(\Psi_{t+1} | \mathcal{F}_t) \leq \Psi_t - \alpha^v \exp\left(-\frac{\mu}{d-1}\right) (1-\alpha) + \alpha^{v-1} \left(1 - \exp\left(-\frac{\mu}{d-1}\right)\right) (1-\rho) d C.$$

Therefore we can set μ small enough, so that we guarantee that the right-hand side above is smaller than Ψ_t for any $v \geq 1$, establishing that Ψ_t is a supermartingale. \square

Acknowledgments

We are thankful to Gábor Pete, Artem Sapozhnikov and Laurent Tournier for useful discussions. Lorenzo Taggi is grateful to Volker Betz for giving him the possibility to work on this project.

References

- [1] G. Amir, O. Gurel-Gurevich: *On Fixation of Activated Random Walks*. Electronic communications in Probability, **15**, Art 12, (2009).
- [2] R. Basu, S. Ganguly, C. Hoffman: *Non-fixation of symmetric Activated Random Walk on the line for small sleep rate*. ArXiv: 1508.05677 (2015).
- [3] R. Dickman, L.T. Rolla, V. Sidoravicius: *Activated Random Walkers: Facts, Conjectures and Challenges*. Journal of Statistical Physics, **138**, 126–142, (2010).
- [4] K. Eriksson: *Chip firing games on mutating graphs*. SIAM J. Discrete Math, **9**, 118–128, (1996)
- [5] C. Hoffman and V. Sidoravicius: unpublished (2004).
- [6] L. T. Rolla and V. Sidoravicius: *Absorbing-State Phase Transition for Driven-Dissipative Stochastic Dynamics on \mathbb{Z}* . Inventiones Mathematicae, **188**, 1, 127–150, (2012).
- [7] L. T. Rolla and L. Tournier: *Sustained Activity for Biased Activated Random Walks at Arbitrarily Low Density*. ArXiv: 1507.04732 (2015).
- [8] E. Shellef: *Nonfixation for activated random walk*. Latin American Journal of Probability and Mathematical Statistics (ALEA), **7**, 137–149, (2010).
- [9] V. Sidoravicius and A. Teixeira: *Absorbing-state transitions for Stochastic Sandpiles and Activated Random Walk*. ArXiv: 1412.7098, (2014).
- [10] L. Taggi: *Absorbing-state phase transition in biased activated random walk*. ArXiv: 1403.1986 (2014).